

JUL 07 2006

Application No. 10/717,870
Amendment dated July 7, 2006
Reply to Office Action of June 6, 2006**AMENDMENTS TO THE CLAIMS:**

The listing of claims will replace all prior versions, and listings of claims in the application:

LISTING OF THE CLAIMS

1-5. (Canceled)

6. (Currently Amended) The An XML query rewrite method as set forth in claim 4, wherein the processing an XML query, the XML query rewrite method comprising:
identifying an expression of the XML query that receives a sequence of concatenated XML items as an expression input;

determining of whether a set of items defined by the output of the expression is independent of grouping of items in the expression input comprises: by determining a granularity property according to:

$$SET(EVAL(exp, LET(I))) = SET\left(\underset{i \text{ in } LET(I)}{CONCAT(EVAL(exp, i))}\right)$$

where exp represents the expression, I represents the sequence of concatenated XML items, CONCAT is a concatenation operator, EVAL is an expression evaluation operator, SET is a set operator, LET produces a concatenated sequence of XML items, and wherein the set of items defined by the output of the expression is determined to be independent of grouping of items in the expression input if the granularity property is true; and

transforming the expression input into a transformed data stream input, the transforming being conditioned upon the set of items defined by the output of the expression being independent of grouping of items in the expression input.

7. (Original) The XML query rewrite method as set forth in claim 6, wherein the transforming of the sequence of concatenated XML items into a transformed data stream input comprises:

transforming the sequence of concatenated XML items into a data stream of individual unconcatenated XML items.

Application No. 10/717,870
 Amendment dated July 7, 2006
 Reply to Office Action of June 6, 2006

8. (Original) The XML query rewrite method as set forth in claim 6, further comprising:

determining a duplication property according to:

$\forall i, j$ in $LET(I)$ with $POSITION(LET(I), i) \neq POSITION(LET(I), j)$ holds
 $SET(EVAL(exp, i)) \cap SET(EVAL(exp, j)) = \{\}$, and

determining an ordering dependence property according to:

$\forall i, j$ in $LET(I)$ holds

$POSITION(LET(I), i) < POSITION(LET(I), j)$

$\Rightarrow \neg \exists e_i$ in $EVAL(exp, i), e_j$ in $EVAL(exp, j)$ such that $e_i \gg e_j$,

wherein POSITION returns ordinal position of an element within a sequence, and the transforming is further conditioned upon the duplication property being true and upon the ordering dependence property being true.

9-11. (Canceled)

12. (Currently Amended) The An XML query rewrite method as set forth in claim 1, wherein the processing an XML query, the XML query rewrite method comprising:

identifying an expression of the XML query that receives an expression input;

determining of whether a set of items defined by the output of the expression is independent of a grouping of items in the expression input comprises: by determining a granularity property including at least:

$$SET\left(\bigcup_{p \in P(I)} CONCAT(EVAL(exp, p))\right) = SET\left(\bigcup_{p \in I} CONCAT(EVAL(exp, p))\right)$$

where exp represents the expression, I represents the expression input; CONCAT is a concatenation operator, EVAL is an expression evaluation operator, P(I) denotes a partitioning of the expression input I, and SET is a set operator, and wherein the set of items defined by the output of the expression is determined to be independent of grouping of items in the expression input if the granularity property is true; and

Application No. 10/717,870
 Amendment dated July 7, 2006
 Reply to Office Action of June 6, 2006

transforming the expression input into a transformed data stream input, the transforming being conditioned upon the set of items defined by the output of the expression being independent of grouping of items in the expression input.

13. (Original) The XML query rewrite method as set forth in claim **12**, wherein the transforming of the expression input into a data stream input comprises:

removing an imposition of a grouping of XML items corresponding to a quantifier that imposes a selected grouping of XML items on the expression input.

14. (Original) The XML query rewrite method as set forth in claim **13**, wherein the XML query is an XQuery, and the removing of an imposition of a grouping of XML items corresponding to a quantifier comprises:

removing an imposition of a grouping of XML items corresponding to a quantifier selected from a group consisting of an XQuery LET quantifier and an XQuery FOR quantifier.

15. (Original) The XML query rewrite method as set forth in claim **12**, further comprising:

determining a duplication property according to:

$$\forall p_i, p_j \text{ in } I \text{ holds } SET(EVAL(exp, p_i)) \cap SET(EVAL(exp, p_j)) = \emptyset$$

and determining an ordering property according to:

$\forall p_i, p_j \text{ in } I$:

p_i occurs before p_j in I

$\Rightarrow \neg \exists e_i \text{ in } EVAL(exp, p_i), e_j \text{ in } EVAL(exp, p_j) \text{ such that } e_i \gg e_j$

wherein the transforming is further conditioned upon the duplication property being true and upon the ordering dependence property being true.

16-20. (Canceled)

Application No. 10/717,870
 Amendment dated July 7, 2006
 Reply to Office Action of June 6, 2006

21. (Currently Amended) The An article of manufacture including a program storage medium storing instructions executable by an associated computer, the instructions embodying an XML query compilation processor as set forth in claim 18, wherein the comprising:

an execution compiler transforming an XML query into an executable XML query plan; and

a query rewrite processor is performing query transformations on the XML query, said query transformations including transforming an expression input received by an expression that produces an output stream of individual XML items conditional upon a set of items defined by the output stream of individual XML items being independent of grouping of items in the expression input and being adapted to perform the expression input transformation by:

determining a granularity property according to:

$$SET(EVAL(exp, LET(I))) = SET\left(\underset{i \in LET(I)}{CONCAT}(EVAL(exp, i))\right)$$

where I represents the expression input, SET is a set operator, CONCAT is a concatenation operator, EVAL is an expression evaluation operator, LET concatenates XML items of expression input I into a sequence, and exp represents the expression receiving the expression input; and transforming the expression input into a transformed data stream input, the transforming being conditioned upon the granularity property being true.

22. (Original) The XML query compilation processor as set forth in claim 21, wherein the query rewrite processor is adapted to perform the expression input transformation by further:

determining a duplication property for each consumer of the output stream of individual XML items indicating whether duplicates in the output stream of individual XML items are problematic for that consumer; and

determining an ordering dependence property for each consumer indicating whether

Application No. 10/717,870
Amendment dated July 7, 2006
Reply to Office Action of June 6, 2006

a selected ordering is required for that consumer; and

applying one or more corrections including at least one of:

removing duplicates from the output stream of individual XML items
flowed to a consumer for which said duplicates are problematic, and
sorting the output stream of individual XML items into a selected
ordering required for a consumer.

23. (Original) The XML query compilation processor as set forth in claim 21, wherein the XML query compilation processor processes XQuery XML queries, the expression input is a concatenated sequence input corresponding to an XQuery LET clause, and the query rewrite processor is adapted to perform the expression input transformation by further:

transforming the concatenated sequence input into a data stream of individual unconcatenated XML items corresponding to an XQuery FOR clause.

24. (Currently Amended) ~~The~~ An article of manufacture including a program storage medium storing instructions executable by an associated computer, the instructions embodying an XML query compilation processor as set forth in claim 18, wherein the comprising:

an execution compiler transforming an XML query into an executable XML query plan; and

a query rewrite processor is performing query transformations on the XML query, said query transformations including transforming an expression input received by an expression that produces an output stream of individual XML items conditional upon a set of items defined by the output stream of individual XML items being independent of grouping of items in the expression input and being adapted to perform the transforming of an expression input transformation by:

determining a granularity property including sub-properties (i), (ii), and (iii) according to:

Application No. 10/717,870
 Amendment dated July 7, 2006
 Reply to Office Action of June 6, 2006

- (i) $SET\left(\underset{p \in P(I)}{CONCAT(EVAL(exp, p))}\right) = SET\left(\underset{p \in I}{CONCAT(EVAL(exp, p))}\right)$.
- (ii) $\underset{p \in P(I)}{CONCAT(EVAL(exp, p))}$ is duplicate-free, and
- (iii) the XML items in $\underset{p \in P(I)}{CONCAT(EVAL(exp, p))}$ are in the selected ordering,

where I represents the expression input, P(I) is a partition operator, SET is a set operator, CONCAT is a concatenation operator, EVAL is an expression evaluation operator, exp represents the expression receiving the expression input, sub-property (ii) being included conditional upon a duplicate-free output stream being required, and sub-property (iii) being included conditional upon a selected ordering of the output stream being required; and removing an imposition of a grouping of XML items on the expression input, the removing being conditioned upon the granularity property meeting a selected criterion.

25-27. (Canceled)

28. (Currently Amended) The ~~An~~ article of manufacture as set forth in claim 25, wherein the including a program storage medium encoding instructions executable by an associated computer, said instructions embodying a query rewrite method comprising: identifying an expression of an XML query having an input data stream grouped by a first quantifier and an output data stream grouped by a FOR quantifier;

computing of at least one evaluation property including properties P1R, P2R, and P3R based at least on the expression and the first quantifier; and

the performing of a selected quantifier transformation conditional upon the computed evaluation property comprise: computing properties P1R, P2R, P3R; and by transforming the selected quantifier into a REG quantifier conditional upon the properties P1R, P2R, P3R being satisfied.

29. (Original) The article of manufacture as set forth in claim 28, wherein the performing of a selected quantifier transformation conditional upon the computed evaluation property further comprises:

Application No. 10/717,870
Amendment dated July 7, 2006
Reply to Office Action of June 6, 2006

transforming the selected quantifier into a REG quantifier conditional upon the property P1R being satisfied; and

performing a de-duplication operation conditional upon the property P2R not being satisfied; and

performing a sorting operation conditional upon the property P3R not being satisfied.

30. (Currently Amended) The An article of manufacture as set forth in claim 25, wherein the including a program storage medium encoding instructions executable by an associated computer, said instructions embodying a query rewrite method comprising:

identifying an expression of an XML query having an input data stream grouped by a first quantifier and an output data stream grouped by a FOR quantifier;

computing of at least one evaluation property including properties P1R, P2R, and P3R based at least on the expression and the first quantifier; and

the performing of a selected quantifier transformation conditional upon the computed evaluation property further comprise: computing properties P1F, P2F, P3F; and by transforming the selected quantifier into a FOR quantifier conditional upon the properties P1F, P2F, P3F being satisfied.

31. (Original) The article of manufacture as set forth in claim 30, wherein the performing of a selected quantifier transformation conditional upon the computed evaluation property further comprises:

transforming the selected quantifier into a FOR quantifier conditional upon the property P1F being satisfied; and

performing a de-duplication operation conditional upon the property P2F not being satisfied; and

performing a sorting operation conditional upon the property P3F not being satisfied.

32-34. (Canceled)